CEESA Meets Machine Learning: A Constant Elasticity Earth Similarity Approach to Habitability and Classification of Exoplanets

Suryoday Basak[a], Snehanshu Saha[a], Kakoli Bora[b], Abhijit Jeremiel Theophilus[a], Margarita Safonova[c], Jayant Murthy[d], Gouri Deshpande[f], Surbhi Agrawal[a]

[a]*Authors are affiliated with the Department of Computer Science and Engineering, and Center for Applied Mathematical Modeling and Simulation (CAMMS), PESIT South Campus, Bangalore, India.*
[b]*Author is affiliated with the Department of Information Science and Engineering, and Center for Applied Mathematical Modeling and Simulation (CAMMS), PESIT South Campus, Bangalore, India.*
[c]*Author is affiliated with Birla Institute of Fundamental Research, Bangalore, India.*
[d]*Author is affiliated with Indian Institute of Astrophysics, Bangalore, India.*
[e]*Author is affiliated with the Department of EECS, University of Calgary, Canada.*
[f]

Abstract

We examine the existing metrics of habitability and classification schemes of extrasolar planets and an provide an exposition of the use of machine learning to estimate habitability and to automate the process of classification of exoplanets respectively. The data used for the analysis is got from Planetary Habitability Laboratory's (University of Puerto Rico) Habitable Exoplanets Catalog (HEC). The catalog exhibits significant imbalance, which are dealt with in a twofold manner for the task of classification: first, by iterative artificial balancing, and second, by artificial data augmentation. In either case, the efficacy of classification algorithms are analyzed and related to the structure and distribution of the data.

Keywords: exoplanets, machine learning, artificial balancing, data augmentation

1. Introduction

The rate of discovery of extrasolar planets (exoplanets) is rapidly increasing. The thought of planets existing other than Earth which could possibly harbor life is intriguing and has captured the imagination of scientists for centuries. In the last decade, thousands of planets were discovered in our galaxy alone. The inference is that stars with planets are a rule rather than exception, with estimates of the actual number of planets exceeding the number of stars in our galaxy by orders of magnitude **what orders of magnitude?**(Strigari et al., 2012). Led by the NASA Kepler Mission (Batalha, 2014), around **3416 planets have been confirmed**, and 4900+ celestial objects remain as candidates that are yet to be confirmed as planets. The discovery and characterization of exoplanets require both extremely accurate instrumentation and sophisticated statistical methods in order to extract the weak planetary signals from the dominant starlight or very large samples. Characterization of Kepler's different planets is important to judge their habitability (Swift et al., 2013). Detailed modeling of planetary signals to extract information of the orbital or atmospheric properties is even more challenging. Moreover, inferring the properties of underlying planet populations from biased or incomplete samples is a challenge.

Upon appreciating the increasing rate of discovery of exoplanets (and with the scheduled release of the James Webb Space Telescope in 2019), it can be expected that the amount of data samples of exoplanets will reach the scale of a big-data problem (much like the the volume of samples collected by SDSS **- rephrase**). In this light, it is important to explore the current classification schemes and to devise methods which can automatically discover meaningful patterns in data and classify them. As no one single parameter would suffice as the sole criteria for habitability, we explore methods which

take into consideration multiple observable characteristics of exoplanets. For example, the presence of water may increase the likelihood for an exoplanet to be a potential habitable candidate (Irwin et al., 2014); if a planet resides in the CHZ of their parent star, it is considered to be a potential candidate for being habitable as the atmospheric conditions in these zones are more likely to support life (Kaltenegger et al., 2011), but in either case, the habitability cannot be affirmed until other parameters are collectively considered: these include, in addition to the ones mentioned, the mean surface temperature, the mass of the planet, radius, etc.

This paper explores existing classification schemes from a machine learning point of view. Thermal categories of planets are primarily considered as habitability classes and data set used contains information about the planetary observables such as mass, temperature, density, etc. A lot of the features are derived, however, they bolster the identity of the planets and prepare a stronger case for classification. These observable parameters can be used as features in a machine learning algorithm. If class labels can be based on subjective examination of all the parameters of an exoplanet, then machine learning can be an effective tool in building a classification scheme. In a parallel exercise to the classification scheme, we also develop a method which does not require target class labels but finds an optimal convex combination of the observables, which is a *habitability score*.

2. Problem Statement

Considering the purview of the complexity of assessing the habitability of exoplanets, there is no way to definitively come to a conclusion regarding habitability classes, types, etc. at this point of time. Hence, it is imperative to explore different methods that can be proved mathematically and whose physical interpretations can be strongly justified. We explore machine learning based classifiers and mathematical models (as metrics) for classification and habitability assessment, respectively, of newly discovered exoplanets. Our proposed methods try to reconcile computational methods, algorithmic learning, and mathematical modeling for determining how habitable an exoplanet might be. The outcome of the all the models may be used as indicators while looking for new habitable worlds.

In previous work, metrics have been developed to estimate habitability or *earth similarity* such as Earth Similarity Index (ESI) (Schulze-Makuch et al., 2011), Biological Complexity Index (BCI) (Irwin et al., 2014), Planetary Habitability Index (PHI) (Schulze-Makuch et al., 2011), and Cobb-Douglas Habitability Score (CDHS) (Bora et al., 2016). $<--$ **perhaps we should elaborate on these, with a single line for each metric based on their working**

Our principal contribution is to propose an integrated approach to habitability classification. Although it may appear as an extension to our previous effort **cite CD-HPF, ProxB**, the manuscript is far beyond an incremental approach....**TO BE CONTINUED..** $<---$ **I think that we should be moderate about how this paper is different because physicists don't like such things; besides, a lot of the papers by a group on the same topic are similar**

3. Justification of the Methodology and Motivation

As per the classification scheme of PHL, exoplanets are classified into classes of habitability based on their surface temperature. While it is reasonable in itself in saying that many factors of life are dependent on temperature, we believe that while trying to assess the habitability of an exoplanet by means of a metric, more than just the temperature should be taken into consideration. Factors such as the radius of a planet, density, escape velocity, eccentricity, etc. are important while determining if a planet can be habitable or not. For example, there might be cases where the surface temperature of a planet is in the habitable range, but the planet is too massive to harbor life like the way it is on Earth. Hence, developing classification schemes based on only one parameter alone is not sufficient. This has been a prime motivation for the development of metrics such as BCI, PHI, and ESI; this in turn

inspired us to explore models that can be used to assess the habitability of exoplanets, which led to the development of CD-HPF. The most significant difference between CD-HPF and the aforementioned habitability metrics is that CD-HPF is inherently adaptive, with the constituent observables in the model being given different levels of importance in different planets; and the overall habitability as indicated by the CD-HPF is a score that is maximized on the constituent variables.

An inherent characteristic of machine learning algorithms is to handle multiple attributes in data. This makes ML an attractive suite of models which can be tried to classify the planets in the existing classification scheme, but while considering more planetary attributes as *features* in the classification algorithms. ML approaches can help uncover what factors the habitability of a planet depends on, and this process is not devoid of human intervention. An algorithm would require an initial input of planetary samples whose potential of harboring life has been recognized, and as the number of exoplanets discovered increases, we would have a machine which could aid us in the process of discovering potentially habitable planets. A common misconception of ML related approaches is that these methods curtain the human understanding of problems they're being applied to. However, such a notion has been pervasive only because methods are often used without a solid justification; the appropriate usage of an ML method is seldom related to the structure of the data it is being applied on, and the context of a problem under consideration.

We have used ML methods and mathematical modeling to develop richer inference from the data of exoplanets which can bolster our understanding of factors that affect habitability in the long run. Instead of leaving the work to a machine, we wish to use our methods to point out to the important aspects of data and use this knowledge gained in tandem to what we know from other physical analyses, to find interesting planetary samples. In this paper, in addition to exploring the efficacy of ML algorithms for classification of exoplanets, we have developed a new metric for habitability, which we call CEESA: the Constant Elasticity of Substitution Earth Similarity Approach. A shortcoming of the CD-HPF is it's multiplicative form; the consequence of this is that while the model is proven to be scalable, it requires all it's constituent variables to have non-zero values. CEESA overcomes this as it's form is inherently additive, and it can work even where there are missing or inappropriate values. While trying to scale up the CD-HPF production function, we faced a bottleneck when we tried to use orbital eccentricity as a feature as eccentricity of a planet is reported as zero if it is naturally zero or if the data is missing from the database. Eccentricity plays a major role to determine the shape of planets. Moreover it controls the climate of a planet **(cite)**. CEESA is hence a generic way of indicating habitability from missing or incomplete data.

The remainder of the paper is organized as follows:

4. Data and the Catalog

In the data set, there exists a huge bias of samples towards the non-habitable category of planets; we propose two distinct methods to address this. The first method is by iterative artificial balancing, as explained in Section (*insert number here*) and the second is by artificial data augmentation as explained in Section (*insert number here*). An exploratory classification is performed on the original data set; the algorithms which perform the best are also tried on an artificially augmented data set.

4.1. Classes and Features in the Dataset

PHL-HEC has been derived from the Hipparcose catalog which contains 118,219 stars. It has been created from the Hipparcose Catalog by examining the information on distances, stellar variability, multiplicity, kinematics, and spectral classification for the stars contained therein. In this study, the HEC has been used because it provides an expanded target list for use in the search for extraterrestrial intelligence by Project Phoenix of the SETI Institute. The HEC data set consists of a total of 68 features of about 3500 confirmed exoplanets (at the time of writing of this paper). The reason behind selecting

the HEC as the source of data is that it combines measured and modeled parameters from various sources. **Hence, it provides a good metric for visualization and statistical analysis (Méndez, 2011). - rephrase this** Statistical machine learning approaches have not been applied on this data set to the best of our knowledge, providing good reasons to explore and exploit accuracy of different machine learning algorithms.

The HEC dataset possesses 13 categorical features and 55 continuous features. There are three classes in the dataset, namely non-habitable, mesoplanets, and psychroplanets on which the ML methods have been tried (there do exist other classes in the dataset on which the methods cannot be tried the reasons for **which has been explained in Section...**. These three labels or classes or types of planets (for the purpose of classification) can be defined on the basis of their thermal properties as follows:

1. **Mesoplanets** (Asimov Issac, 1998) <— **change asimov citation**: **The planetary bodies whose sizes lie between Mercury and Ceres falls under this category (smaller than Mercury and larger than Ceres).** These are also referred to as M-planets (Méndez, 2011). These planets have mean global surface temperature between 0°C to 50°C, a necessary condition for complex terrestrial life. These are generally referred as Earth-like planets.
2. **Psychroplanets** (Méndez, 2011): These planets have mean global surface temperature between -50°C to 0°C. Hence, the temperature is colder than optimal for sustenance of terrestrial life.
3. **Non-Habitable**: Planets other than mesoplanets and psychroplanets do not have thermal properties required to sustain life.

The catalog includes features like atmospheric type, mass, radius, surface temperature, escape velocity, earth's similarity index, flux, orbital velocity etc. Online data source for the current work is available at http://phl.upr.edu/projects/habitable-exoplanets-catalog/data/database.

The data flow diagram of the entire system is depicted in As a first step, data from the HEC catalog is pre-processed (the authors have tried to tackle the missing values by taking mean for continuous valued attribute and mode for categorical attributes). Certain attributes from the database namely P.NameKepler (planet's name), Sname HD and Sname Hid (name of parent star), S.constellation (name of constellation), Stype (type of parent star), P.SPH (planet standard primary habitability), P.interior ESI (interior earth similarity index), P.surface ESI (surface earth similarity index), P.disc method (method of discovery of planet), P.disc year (year of discovery of planet), P. Max Mass, P. Min Mass, P.inclination and P.Hab Moon (flag indicating planet's potential as a habitable exomoons) were removed as these attributes do not contribute to the nature of classification of habitability of a planet. Interior ESI and surface ESI, however, together contribute to habitability, but since the data set directly provides P.ESI, these two features were neglected. Following this, classification algorithms were applied on the processed data set. **In all, 51 features are used**.

4.2. Data Imbalance in the Dataset

The primary problem with the catalog from a data-analytic point of view is that of data imbalance. The number of non-habitable samples in the data is over a thousand times the number of samples of the other two classes. This makes the dataset challenging to handle for a machine learning exploration. If machine learning classifiers are deployed on the data without handling the inherent data bias, the results thus achieved will also be extremely biased due to the presence of a dominating class, and the classifiers thus developed will be inappropriate to classify samples at scale. In order to overcome this shortcoming, two separate methods have been tried. In the first method, we have tried to handle the effects of data bias by means of artificially balancing the data for each iteration of the experiment; here, we create smaller datasets with a balanced number of samples across all the classes iteratively to build many classifiers. This is an *ensemble* approach, whose results are appropriate as compared to deploying a classifier on the bulk of the data. In the second method, we have tried to oversample the

minority classes in order to balance the datasets for the experiments; here, the samples in the minority classes are increased, thus balancing the number of samples across all classes.

4.3. Eccentricity Estimation

To calculate missing values of eccentricity of planets in the given exoplanet catalog.

4.3.1. Data exploration

The data set consists of 68 features spanning 3664 data points. Initially, it was discovered that there were 6 features that consisted of the names of the exoplanets, discovery methods, and discovery years which didn't play any role in the missing value imputation. Further analysis of the data set revealed that over 60% of the eccentricity values were missing, and the values of eccentricity ranges from -2,000 to +26,000.

4.3.2. Research about Eccentricity

The orbital eccentricity of an astronomical object is a parameter that determines the amount by which its orbit around another body deviates from a perfect circle. A value of 0 is a circular orbit, values between 0 and 1 form an elliptical orbit, 1 is a parabolic escape orbit, and greater than 1 is a hyperbola.
This observation revealed that the data points that were marked as negative in the dataset did were recorded wrong. A negative eccentricity implies that the Aphelion and the Perihelion were flipped which does not make logical sense.
Research about eccentricity revealed the following:

- Number of planets in the system
- Mass of the other planets in the solar system planets
- Distance from the sun and other heavy bodies(asteroids etc)
- An indirect relationship was found with the composition and atmosphere. Eccentricity of a planet affects the climate, atmosphere, and the composition of a planet to a large degree. Hence, by backtracking, eccentricity can be calculated by using these parameters.

4.3.3. Dimensionality reduction

Incremental principal component analysis was used to reduce the number of features. This algorithm assigns weights to every feature. Features like luminosity or number of moons do not play as large a role in computing the eccentricity of a planet as does the mass and number of neighboring planets.

4.3.4. Eccentricity calculation

Regression algorithms were used to calculate the missing values. L1 Lasso regularization gave the highest accuracy of 70%.

- L1 Lasso regularization: Lasso is a regression analysis method that performs both variable selection and regularization in order to enhance the prediction accuracy and interpretability of the statistical model it produces. It was introduced by Robert Tibshirani in 1996 based on Leo BreimanâĂŹs nonnegative garrote. Lasso was originally formulated for least squares models and this simple case reveals a substantial amount about the behavior of the estimator, including its

relationship to ridge regression and best subset selection and the connections between lasso coefficient estimates and so-called soft thresholding.

Consider a sample consisting of N cases, each of which consists of p covariates and a single outcome. Let y_i be the be the outcome and $x_i := (x_1, x_2, x_3, ...)^T$ be the covariate vector for the i^th case. Then, the objective of Lasso is to solve:

$$\min_{\beta_0,\beta} \{ \frac{1}{N} \sum_{i=1}^{N} (y_i - \beta_0 - x_i^T \beta)^2 \}, subjectto \sum_{j=1}^{p} |\beta_j| \leq t. \quad (1)$$

The following features are used to regress the value of eccentricity:

- Zone Class
- Mass Class
- Atmosphere Class
- Composition Class
- Mass of the planet

L1 Lasso regularization gave an accuracy of 70%.

P. Eccentricity
0.03
0.23
0.03
0.16
0.53
0.15
0.16
0.12
0.03
0.07

Figure 1: Calculated values

4.3.5. Relationship between eccentricity and mass

Relationships between eccentricity and mass:

- Eccentricity Vs. Mass

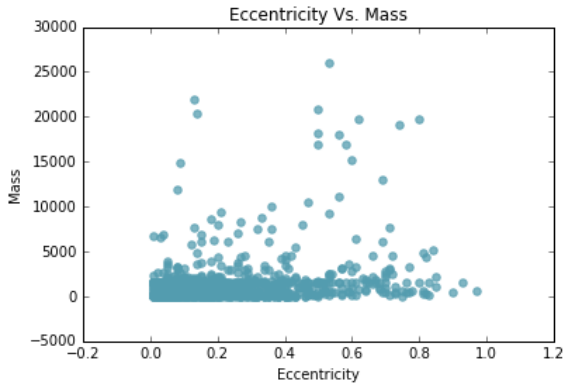


Figure 2: Eccentricity Vs. Mass

4.4. Surface temperature imputation

To calculate missing values of Surface Temperature in the given exoplanet catalog.

4.4.1. Data exploration

The data set consists of 68 features spanning 3664 data points. Initially, it was discovered that there were 6 features that consisted of the names of the exoplanets, discovery methods, and discovery years which didn't play any role in the missing value imputation. Further analysis of the data set revealed that over 30% of the eccentricity values were missing or erroneous. The ones which were recorded wrong were given an arbitrary value of 336.17K.

4.4.2. Research about Surface Temperature

The surface temperature of a terrestrial planet is determined by how much energy the planet receives from the Sun and how quickly it radiates that solar energy back to space.
Tree things can affect the energy flow and therefore, the average global surface temperature. They are the planet's distance from the Sun, the planet's surface reflectivity (albedo), and the planet's atmosphere.
Research about Surface Temperature revealed the following:

- Planets closer to the Sun receive more solar energy by an amount that depends on their distance squared.
- With more solar energy flowing to the closer planets, they must be hotter to re-radiate that energy back out to space. The amount of solar energy reflected immediately out to space is determined by the material on the planet's surface or clouds in the atmosphere. The fraction of sunlight that is reflected from an object is the albedo. If the albedo is closer to 1 (100% reflectivity), the planet does not need to be as hot to have its outflow of energy balance the inflow of solar energy. Darker objects absorb more solar energy and, therefore, they need to heat up more to re-radiate that energy back out to balance the inflow of solar energy.
- Atmosphere of the Planet
- Rate of energy energy absorbed by the planet

4.4.3. Dimensionality reduction and Surface Temperature Calculation

Sparse principal component analysis (sparse PCA) is a specialised technique used in statistical analysis and, in particular, in the analysis of multivariate data sets. It extends the classic method of principal component analysis (PCA) for the reduction of dimensionality of data by adding sparsity constraint on the input variables.

Ordinary principal component analysis (PCA) uses a vector space transform to reduce multidimensional data sets to lower dimensions. It finds linear combinations of input variables, and transforms them into new variables (called principal components) that correspond to directions of maximal variance in the data. The number of new variables created by these linear combinations is usually much lower than the number of input variables in the original dataset, while still explaining most of the variance present in the data.

A particular disadvantage of ordinary PCA is that the principal components are usually linear combinations of all input variables. Sparse PCA overcomes this disadvantage by finding linear combinations that contain just a few input variables.

Consider a data matrix, X, where each of the p columns represent an input variable, and each of the n rows represents an independent sample from data population. One assumes each column of X has mean zero, otherwise one can subtract column-wise mean from each element of X. Let

$$\Sigma = (\frac{1}{n-1})X^T X \tag{2}$$

be the empirical covariance matrix of X, which has dimension $p * p$. Given an integer k with $1 \leq k \leq p$, the sparse PCA problem can be formulated as maximizing the variance along a direction while constraining its cardinality: max

$$v^T \Sigma v \tag{3}$$

subject to

$$||v||_2 = 1 ||v||_0 \leq k. \tag{4}$$

After finding the optimal solution v , one deflates Σ to obtain a new matrix

$$\Sigma_1 = \Sigma - (v^T \Sigma) v v^T, \tag{5}$$

and iterate this process to obtain further principal components.

The following features are used to regress the value of Surface Temperature:

- Distance from the nearest sun
- Mass Class
- Atmosphere Class
- Composition Class
- Mass of the planet
- Density of the planet

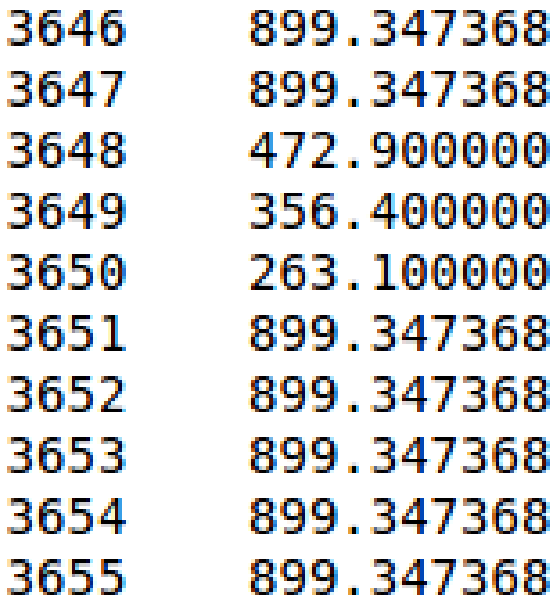


Figure 3: Calculated values

5. Machine Learning Methodology for Exoplanet Classification

5.1. Handling Data Bias by Artificial Balancing

In this method of addressing data bias in the dataset, the dataset was divided into smaller datasets with the number of samples in each class being equal to the number of samples in the psychroplanet class (**INSERT NUMBER HERE**), as the number of psychroplanet samples in the dataset is the least. This is after excluding the classes of thermoplanet and hypopsychropanet, whose samples are too less for a machine learning exploration (**insert numbers**). The classifiers are built and tested for each balanced subset of the data. **X number** of such subsets are built and the classifiers are tested on them.

This iterative procedure of building smaller and balanced datasets is used to smooth out any effects of bias that arise due to imbalance. **In general, it is similar to the process of bootstrap aggregation** – we can try to prove this!. Building a classifier directly on the imbalanced dataset is not proper methodology, the evidence of which is provided in the **supplementary material**.

5.2. Artificial Data Augmentation

The reliability of the semi-automatic process depends on the efficacy of the classifiers if the data set grew rapidly with many entities. Since the required data are not naturally available, the authors have simulated a data generation process, albeit briefly, and performed classification experiments on the artificially generated data. The strategy has a two-fold objective: to devise a preemptive measure to check scalability of the classifiers, and to tackle classes of exoplanets with insufficient data. We elaborate the concept, theory, and model in this subsection and establish the equivalence of both premises.

The two different methods explored for the simulation are:

1. By assuming a Poisson distribution in the data.
2. By estimating an empirical distribution from the data.

The authors would insist on the usage of empirical distribution estimation over the assumption of a distribution. Nonetheless, the first method paved way to the next, more robust method. The naturally occurring data points are relatively less in order to describe the distribution of data by a known distribution (such as Poisson, or Gaussian). If a known distribution is estimated using this data, chances are that the distribution thus determined is not representative of the actual density of the data. As this fact is almost impossible to establish at this point in time, two separate methods of synthesizing data have been developed and implemented to gauge the efficacy of ML algorithms.

5.2.1. Generating Data by Assuming a Distribution

The challenge with artificially oversampling data in PHL-EC is that the original data available is too less to estimate a reliable probability distribution which is satisfactorily representative of the probability density of the naturally occurring data. For this, a *bounding mechanism* should be used so that while augmenting the data set artificially, the values of each feature or observable does not exceed the physical limits of the respective observable, and the physical limits are analyzed from the naturally occurring data.

For this purpose, we use a hybrid of SVM and K-NN to set the limits for the observables. The steps in the SVM-KNN algorithm are summarized below:

Step 1: The best boundary between the psychroplanets and mesoplanets are found using SVM with a linear kernel.

Step 2: By analyzing the distribution of either class, data points are artificially created.

Step 3: Using the boundary determined in Step 1, an artificial data point is analyzed to determine if it satisfies the boundary conditions: if a data point generated for one class falls within the boundary of the respective class, the data point is kept in it's labeled class in the artificial data set.

Step 4: If a data point crosses the boundary of its respective class, then a K-NN based verification is applied. If 3 out of the nearest 5 neighbors belongs to the class to which the data point is supposed to belong, then the data point is kept in the artificially augmented data set.

Step 5: If the conditions in Steps 3 and 4 both fail, then the respective data point's class label is changed so that it belongs to the class whose properties it corresponds to better.

Step 6: Steps 3, 4 and 5 are repeated for all the artificial data points generated, in sequence.

Here, K-NN and SVM are used, along with density estimation, to rectify the class-belongingness (class labels) of artificially generated random samples. If an artificially generated random sample is generated such that it does not conform to the general properties of the respective class (which can be either mesoplanets or psychroplanets), the class label of the respective sample is simply changed such that it may belong to the class of habitability whose properties it exhibits better. The strength of using this as a rectification mechanism lies in the fact that artificially generated points which are near the boundary of the classes stand a chance to be rectified so that they might belong to the class they better represent. Moreover, due to the density estimation, points can be generated over an entire region of the feature space, rather than augmenting based on individual samples. **This aspect of the simulation is the cornerstone of the novelty of this approach: in comparison to existing approaches as SMOTE (Synthetic Minority Oversampling Technique) (?), the oversampling does not depend on individual samples in the data.**

In this method, we selected the mean surface temperature as the primary discriminating feature since it emerged as the most important feature amongst the classes in the catalog. From Figure 4, it can be understood that the mean surface temperature provides the best discrimination between classes; for different classes of planets, it falls in different intervals of values (Méndez, 2011). It is

the best distinguishing feature of exoplanets, and is a better distinguishing aspect between different classes of planets than mass, escape velocity, density, etc. The mean surface temperature was fit to a Poisson distribution; the vector of remaining features was randomly mapped to these randomly generated values of S. Temp. The resulting vectors of artificial samples may be considered to be a vector $S = (Temp_{Surface}, X)$, where X is any naturally occurring sample in the PHL-EC data set without its corresponding value of the surface temperature. The set of the pairs (S, c) thus becomes an entire artificial catalog, where c is the class label.

5.2.2. Generating Data by Analyzing the Distribution of Existing Data Empirically: Window Estimation Approach

In this method of synthesizing data samples, the density of the data distribution is approximated by a numeric mathematical model, instead of relying on an established analytical model (such as Poisson, or Gaussian distributions). The justification of trying this method is that some classes in the data set are extremely sporadic. The process outlined for this estimation of the population density function was described independently by **?** and **?** and is termed Kernel Density Estimation (KDE). KDE, as a non-parametric technique, requires no assumptions on the structure of the data and further, with slight alterations to the kernel function, may also be extended to multivariate random variables. The method is described below (**I think it should go into the appendix**).

Let $X = x_1, x_2, \ldots, x_n$ be a sequence of independent and identically distributed multivariate random variables having d dimensions. The window function used is a variation of the uniform kernel defined on the set R^d as follows:

$$\phi(u) = \begin{cases} 1 & u_j \leq \frac{1}{2} \quad \forall j \in \{1, 2, \ldots, d\} \\ 0 & otherwise \end{cases} \tag{6}$$

Additionally, another parameter, the edge length vector $h = \{h_1, h_2, \ldots h_d\}$, is defined, where each component of h is set on a heuristic that considers the values of the corresponding feature in the original data. If f_j is the column vector representing some feature $j \in X$ and

$$\begin{aligned} l_j &= min\{(a-b)^2 \quad \forall\, a, b \in f_j\} \\ u_j &= max\{(a-b)^2 \quad \forall\, a, b \in f_j\}, \end{aligned} \tag{7}$$

the edge length h_j is given by,

$$h_j = c\left(\frac{u_j + 2l_j}{3}\right) \tag{8}$$

where c is a scale factor.

Let $x' \in R^d$ be a random variable at which the density needs to be estimated. For the estimate, another vector u is generated whose elements are given by:

$$u_j = \frac{x_j{}' - x_{ij}}{h_j} \qquad \forall j \in \{1, 2, \ldots, d\} \tag{9}$$

The density estimate is then given by the following equation:

$$p(x') = \frac{1}{n \prod_{i=1}^{d} h_i} \sum_{i=1}^{n} \phi(u) \tag{10}$$

Traditionally, random numbers are generated from an analytic density function by inversion sampling. However, this would not work on a numeric density function unless the quantile function is numerically approximated by the density function. In order to avoid this, a form of rejection sampling has been used.

Let r be a d-dimensional random vector with each component drawn from a uniform distribution between the minimum and maximum value of that component in the original data. Once the density, $p(r)$ is estimated by Equation (10), the probability is approximated to:

$$Pr(r) = p(r) \prod_{j=1}^{d} h_j \tag{11}$$

To either accept or reject the sample r, another random number is generated from a uniform distribution within the range $[0, 1)$. If this number is greater than the probability estimated by Equation (11), then the sample is accepted. Otherwise, it is rejected.

We used data synthesis using KDE and rejection sampling (refer to **Supplementary file ??** for visual details to generate a synthetic data set. For the PHL-EC catalog, synthetic data was generated for the mesoplanet and psychroplanet classes by estimating their density by Equation (10) taking $c = 4$ for mesoplanets and $c = 3$ for psychroplanets. 1000 samples were then generated for each class using rejection sampling on the density estimate. In this method, the bounding mechanism was not used and the samples were drawn out of the estimated density. Here, the top 16 features (top 85% of the features by importance, Table **??**) were considered to estimate the probability density, and hence the boundary between the two classes using SVM was not constructed. The values of the remaining features were copied from the naturally occurring data points and shuffled between the artificially augmented data points in the same way as in the method described in Section **??**). The advantage of using this method is that it may be used to estimate a distribution which resembles more closely the actual distribution of the data. However, this process is more complex and takes a longer time to execute. **Nonetheless, the authors would assert this as a method of synthetic oversampling than the method described in Section ?? as it is inherently unassuming and can accommodate distributions in data which are otherwise difficult to describe using the commonly used methods for describing the density of data.**

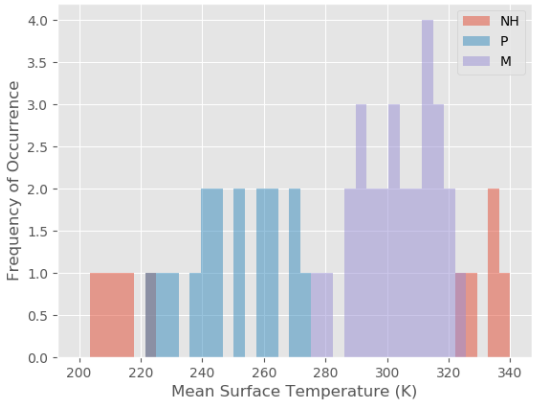
5.3. Justification for Using Machine Learning Models

In Figure 4, we have included the distribution of samples from the various classes of habitability in an attempt to justify the usage of machine learning models to classify new samples in the data. The different classes of habitability are based on the mean surface temperature of exoplanets. However, there exist some overlaps between the classes, as shown in Figure 4(a). If the catalog grows at scale, the overlap between classes may lead to confusion in the class belongingness of data samples. Here, different ML methods along with their properties are explored in order to tackle issues inherent to the PHL-HEC data set.

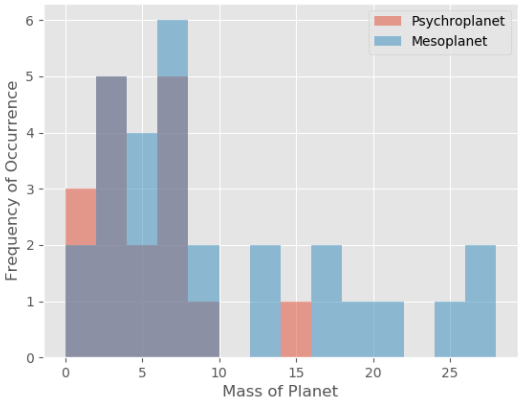
5.4. Classification Algorithms Used

Scikit-learn (**?**) was used to perform these experiments. A brief overview of the classifiers used and their respective settings (in Scikit-learn) are provided below:

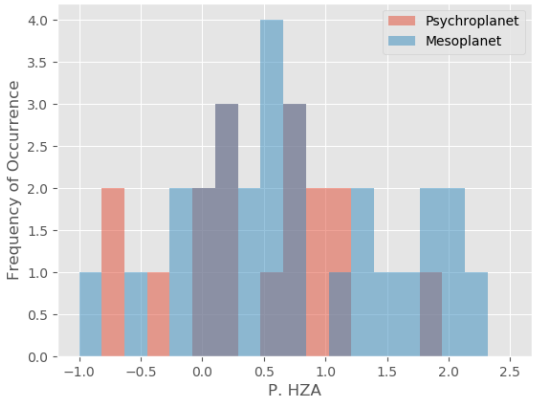
1. *Gaussian Naïve Bayes*: Naïve Bayes classifier is based on Bayes' theorem. It can perform the classification of an arbitrary number of independent variables and is often used when data has many attributes. Consequently, this method is of interest since the catalog used, HEC, has a large number of attributes. The data to be classified may be either *categorical*, such as P.Zone Class or P.Mass Class, or *numerical*, such as P.Gravity or P.Density. A small amount of training data is sufficient to estimate necessary parameters (**?**). The method assumes independent distribution for attributes and thus estimates class conditional probability. It evaluates the classification labels of test data based on class conditional probabilities with class apriori probabilities, class count, mean and variance, calculated from the samples in the dataset.



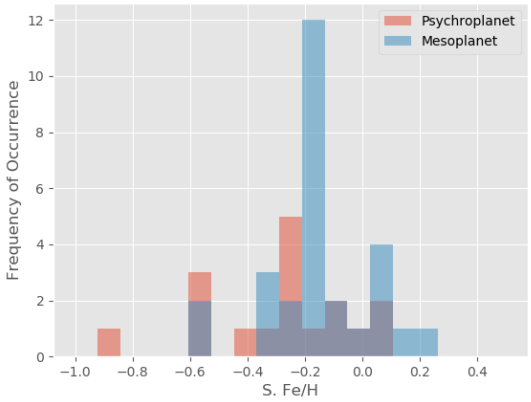
(a) Distribution of samples of non-habitable, psychroplanets and mesoplanets against mean surface temperature.



(b) Distribution of samples of psychroplanets and mesoplanets against mass of the planet.



(c) Distribution of samples of psychroplanets and mesoplanets against HZA the planet.



(d) Distribution of samples of psychroplanets and mesoplanets against Fe/H of the parent star.

Figure 4: Distribution of planets of different classes against different different planetary parameters. The best separation of the different classes is achieved based on mean surface temperature. The other classifications based on mass, atmosphere (HZA) and the ratio of iron to hydrogen (Fe/H) do not result in clearer boundaries between different classes. Though the classification scheme is based on the temperature, the distinction of planets into different classes is not strict and there exist some overlaps.

2. *K-Nearest Neighbor*: It is an instance-based classifier that compares a test instance with the data stored in memory (**?**). KNN uses a suitable distance or similarity function (such as Euclidean, or Mahalanobis distance) and relates new problem instances to the existing ones in the memory. K neighbors are located and majority vote outcome, based on the most occurring class in the K nearest neighbors, decides the class of the test sample. In the current work, the KNN classifier was used with the K value being set to 3 while the weights for all the distances were assigned uniform values.
3. *Support vector machines*: SVM classifiers are effective for binary class discrimination (**?**) **CITE CORTES-VAPNIK**. The basic formulation is designed for the linear classification problem; the algorithm yields an optimal hyperplane i.e. one that maintains the largest minimum distance from all the training data, defined as the margin for separating different classes. In the case where the data corresponding to different classes are inseparable, a penalty term is introduced for training samples which fall on the wrong side of the hyperplane. It can also perform non-linear classification by using the *kernel trick*, which involves the transformation of the data into a different higher dimensional space, where a separating hyperplane may be found. In the current work, the classifier was used with a penalty parameter C of the error term, initialized to default 1.0, while the kernels tried were those of linear, and radial basis function (RBF) (**?**), and the **gamma parameter (kernel coefficient) was assigned to 0.0, the coefficient of the kernel was set to 0.0 as well.**
4. *Decision Trees* and *Random Forests*: Decision trees are tree data structures which are built by using a splitting criterion (Gini impurity or information gain). A tree structure is built by recursively partitioning the feature space using the splitting criteria. Such classification schemes are useful in cases where the data from different classes are inseparable. A *random forest* is an ensemble of decision trees, where each DT is built on a subspace of the dataset (**CITE CITE CITE**). In the case of a random forest, each tree *votes* towards the classification of a test sample; for a test sample, the highest voted class is taken as the predicted class.
5. *Gradient Boosted Trees*: In GBMs, the data in each node of a tree are modeled using a set of *regressor functions*. This additive learning is usually more effective in modeling trends in the data as compared to the simpler splitting criteria employed in DTs and RFs. As a result, the number of trees that are effectively required to perform a classification comparable to a similar random forest with a Gini splitting criteria is significantly less. XGBoost (cite Chen) is a new framework which optimizes the trees which are being built by parallelization: this framework has made the usage of GBMs more practical.

In order to understand if an exoplanet is potentially habitable or not, multiple factors need to be considered. The CEESA and classification models do not serve the purpose of self-contained methods for understanding habitability. Rather, they should be considered as indicators which can provide an augmented perspective on the properties of an exoplanet.

5.5. Results of Machine Learning Algorithms

The results of the experiments have been divided into three parts. The first part is without any bias handling prior to developing the classifiers, and the second and third parts are with artificial data balancing and artificial data augmentation, respectively. This serves as an exposition of the effects of data bias on the results and how the bias has been handled.

5.5.1. Results without artificial data balancing
5.5.2. Results with artificial data balancing
5.5.3. Results with artificial data augmentation

6. CEESA: A New Metric for Evaluating the Habitability of an Exoplanet

The CD-HPF (Bora et al., 2016) is a novel indicator of habitability of an exoplanet. However, the disadvantage of the model is that it is in a multiplicative form. Hence, if the value of an observable is reported as zero, the Cobb-Douglas habitability score (CDHS) of that planet becomes zero, but that is an invalid CDHS for the exoplanet. In this light, it should be noted that none of the observables currently used in the CD-HPF model can have a zero value (the values of radius, surface temperature, density, and escape velocity, which are used in the CD-HPF model, cannot be zero for any planet!). To overcome the shortcomings of the CD-HPF, we try another model here, the most significant deviation being that of the form: the metric we develop here is in an additive form and hence it can handle naturally occurring or spurious zero values of observables. Any number of input parameters can be added to this model. The properties of Constant Elasticity of Scale (CES) production function motivated us to check the applicability of it in our problem domain.

A production function with constant elasticity of substitution (CES) between the inputs has two major characteristics.

- it is homogeneous of degree one
- it has a constant elasticity of substitution

The general form of the Constant Elasticity of substitution (CES) production function for two inputs is

$$Q(L, K) = (\alpha L^{\rho} + (1 - \alpha)K^{\rho})^{\eta/\rho} \tag{12}$$

where
$Q =$ Quantity of output
$L, K =$ Labor and capital, respectively
$\rho = \frac{s-1}{s}$
$s = \frac{1}{1-\rho}$, Elasticity of substitution
η = a measure of the economies of scale or elasticity of scale
and α = Share parameter

Few definitions -

- **Mathematical Optimization** Optimization is one of the procedures to select the best element from a set of available alternatives in the field of mathematics, computer science, economics, or management science (Hájková & Hurnik, 2007). An optimization problem can be represented in various ways. Below is the representation of an optimization problem. Given a function $f : A \to R$ from a set A to the real numbers R. If an element x_0 in A is such that $f(x_0) \leq f(x)$ for all x in A, this ensures minimization. The case $f(x_0) \geq f(x)$ for all x in A is the specific case of maximization. The optimization technique is particularly useful for modeling the habitability score in our case. In the above formulation, the domain A is called a search space of the function f, CD-HPF in our case, and elements of A are called the candidate solutions, or feasible solutions. The function as defined by us is a utility function, yielding the habitability score CDHS. It is a feasible solution that maximizes the objective function, and is called an optimal solution under the constraints known as Returns to scale.
- **Returns to Scale**: measure the extent of an additional output obtained when all input factors change proportionally (Bora et al., 2016). There are three types of returns to scale, which depend

on the values of elasticity of scale and elasticity of substitution (Elmer, 2017; Hawkin, 2013). The range of the elasticity of substitution, ρ, is between 0 to 1. They are described below:

1. **Constant returns to scale (CRS)**. In this case, a proportional increase in all inputs will increase output by the proportional constant. For example, when we multiply the amount of every input (say, L and K) by the number N, the factor by which output increases is more than N. In CES, $\eta = 1$ for CRS.
2. **Decreasing returns to scale (DRS)**. A proportional increase in all inputs will increase output by less than the proportional constant. For example, when we multiply the amount of every input by the number N, the factor by which output increases is less than N. In CES, $\eta < 1$ for DRS case.
3. **Increasing returns to scale (IRS)**. Here, a proportional increase in all inputs will increase output by more than the proportional constant. For example, when we multiply the amount of every input by a number N, the resulting output is multiplied by N. This phase happens for a negligible period of time and can be considered as a passing phase between IRS and DRS. In CES, $\eta > 1$ for IRS case.

- **Computational Techniques in Optimization** There exist several well-known techniques including Simplex, Newton-like and Interior point-based techniques (Nemirovski and Todd, 2008). One such technique is implemented via MATLAB's optimization toolbox using the function **fmincon**. This function helps find the global optima of a constrained optimization problem which is relevant to the model proposed and implemented by the authors. Illustration of the function and its syntax are provided in Appendix...

The Constant Elasticity Earth Similarity Approach (CEESA) is based on the Constant Elasticity of Scale (CES) production function. Here we considered five parameters to estimate the habitability score of planets, which are: radius, density, surface temperature, escape velocity and Eccentricity. In this production function, the elasticity, ρ, is assumed to be a constant. The CEESA model is shown in equation (13). This function is concave if the value of ρ falls in this range:

$$\begin{aligned} &\rho < 0 \\ &0 < \rho \leq 1, and \\ &1 \leq \rho \end{aligned}$$

and thus a maxima is assured to exist in the range of $0 < \rho \leq 1$. As the values of the constituent parameters across a large sample change over time, the model can adapt to find a value of ρ which will lead the model to find the most habitable planets from a large population.

6.1. ***The Analytical model***

The CEESA production function for more than two inputs can be written as:

$$Y = f(R, D, T_s, V_e, E) = (r.R^{\rho} + d.D^{\rho} + t.T_s^{\rho} + v.V_e^{\rho} + e.E^{\rho})^{\frac{\eta}{\rho}} \tag{13}$$

where, R is radius, D is density, T_s is surface temperature and V_e is escape velocity and E is the eccentricity of an exoplanet, which are given (in the data set), r, d, t, v, and e are the coefficients of radius, density, surface temperature, escape velocity, and eccentricity, respectively, and Y is the target output. The sum of the coefficients (r, d, t, v, and e) should be 1. Y is the habitability score of exoplanets, where the aim is to maximize Y subject to the constraint that the range of ρ value is $0 < \rho \leq 1$ and with the value of ν between 0 and 1.

The optimization can be conceptualized as a cost against the revenue, which is Y. Here, we consider cost to be a linear combination of the values of the features. Hence, the goal is to minimize cost and to maximize profit. The cost function may be written as:

The cost for producing y_{tar} units is:

$$c = w_1 R + w_2 D + w_3 T_s + w_4 V_e + w_5 E \tag{14}$$

where,
w_1, w_2, w_3, w_4 and w_5 are the weights of the inputs radius, density, surface temperature, escape velocity and eccentricity respectively.

And thus, the optimization becomes:

$$\min w_1 R + w_2 D + w_3 T_s + w_4 V_e + w_5 E \text{ subject to } Y \tag{15}$$

The sum of the weights should be 1. The profit function for five parameters is thus:

$$\pi = p \cdot Y - w_1 \cdot x_1 - w_2 \cdot x_2 - w_3 \cdot x_3 - w_4 \cdot x_4 - w_5 \cdot x_5$$

where, p is the price.
We can write the profit function as:

$$\pi = pf\left(R, D, T_s, V_e, E\right) - w_1 R - w_2 D - w_3 T_s - w_4 V_e - w_5 E\,, \tag{16}$$

Profit can be maximized when,

$$p\frac{\partial f}{\partial R} = w_1\,, \quad p\frac{\partial f}{\partial D} = w_2\,, \quad p\frac{\partial f}{\partial T_s} = w_3\,, \quad p\frac{\partial f}{\partial V_e} = w_4\,, \quad p\frac{\partial f}{\partial E} = w_5$$

The habitability score is conceptualized as a profit function (Bora et al., 2016).

6.2. ***Implementation of the Model***

We applied the CES production function to calculate the habitability score of exoplanets. A total of 1644 confirmed rocky exoplanets are taken from the Planetary Habitability Laboratory Exoplanets Catalog (PHL-EC) [1]. The catalog contains observed and estimated stellar and planetary parameters for a total of 3689 (September 2017). For our analyses, we have taken all rocky planets. Otherwise, the train and test samples would become heavily biased towards one particular trend. We have normalized the surface temperatures T_s of exoplanets to the EU, by dividing each of them with Earth's mean surface temperature, 288 K (Bora et al., 2016).

With all the input parameters represented in EU, we are looking for the exoplanets whose CEESA score is close to Earth's CEESA score. For each exoplanet, we obtain the optimal elasticity value and the maximum habitability score using fmincon function. All simulations were conducted using the MATLAB software for the cases of **DRS and CRS** (refer Appendix C).

6.3. Computation of CES Score in DRS and CRS phase

We have computed elasticity values for CES in the DRS and CRS phases using function *fmincon*, a computational optimization technique explained in Appendix C. Table 1 and Table 2 show a sample of computed values. The optimal score for most of the exoplanets for DRS are obtained at $\rho = 0.99$ and for CRS at $\rho = 0.91$.

[1] provided by the Planetary Habitability Laboratory @ UPR Arecibo, accessible at `http://phl.upr.edu/projects/habitable-exoplanets-catalog/data/database`

The strength of this kind of econometric modeling is that it can naturally handle missing data or data points with zero values. The motivation behind attempting to develop metrics for habitability in this manner is to be able to observe trends from incomplete or unavailable data to the best of technological ability and the CEESA model can naturally accomplish this. This model is also scalable and can be extended to accommodate more planetary observables (the proof of this is included in Appendix **A**).

6.4. ***Results of CEESA***

For the sake of brevity, the CEESA scores of some interesting habitable exoplanets are given (sample) in Tables 1 and 2 represent habitability score for $\nu = 1$ and $\nu < 1$ respectively, where the corresponding values of elasticities were found by *fmincon*. The elasticity values were recorded for CRS as $\rho = 0.91$ and for DRS as $\rho = 0.99$. We have cross-checked these planets with the Habitable Exoplanets Catalog and found that they are indeed listed as potentially habitable planets.

Table 1: Potentially habitable exoplanets considering Earth as reference for CRS ($\nu = 1$, $\rho \leq 1$): Outcome of CEESA

Exoplanet	Habitability Score
Earth	0.99
Kepler-186 f	1.15
Proxima Cen b	1.10
TRAPPIST-1 e	0.91
TRAPPIST-1 f	1.02
Ross 128 b	1.14

Table 2: Potentially habitable exoplanets considering Earth as reference for DRS ($\nu < 1$, $\rho \leq 1$): Outcome of CEESA

Exoplanet	Habitability Score
Earth	0.99
Kepler-186 f	0.99
Proxima Cen b	0.99
TRAPPIST-1 e	0.98
TRAPPIST-1 f	0.98
Ross 128 b	1.01

This list of optimistic potentially habitable planets is derived from the PHL-HEC catalog.

7. Discussion and mapping with results from section 5

8. Conclusion and Future Work

The authors provide a manuscript that develops a tool for planetary habitability. The tool is developed using known functions to score habitability combined with planetary features to generate a predictor. That predictor is developed using tree-building approaches or non-ensemble methods to produce a similar outcome.

The concept of developing a classifier based on our growing knowledge of exoplanets is intriguing. There is no reason why such an approach shouldn't work, other than to think that of the large number of possible habitable exoplanets, we have parameters based on only one example that is known to be habitable and in that regard assume that all non-Earth like exoplanets are non-habitable. Our definition of habitability may need to be refined as we find more truly habitable planets.

Future work may involve adapting Computational Intelligence (CI) based ideas. CI is classically defined as neural networks, fuzzy systems, or evolutionary computation or other nature-inspired algorithm approaches. Instead of making use of stochastic gradient ascent to find local/global maxima, evolutionary algorithms can be used to track dynamic functions of the type that allow for the oscillation that the authors instead mitigate with fmincon library. Additionally, we make use of 49 features through XGBoost. And the results suggest that the use of Proxima b for training and remaining samples in the catalog for testing performed well with XGBoost (AUC 1.0) which is surprisingly good.

Therefore, it is reasonable to expect a neural network delivering comparable performance. If XGBoost has AUC >0.99 on training, we would expect even a vanilla feedforward neural network trained with backpropagation would have similar accuracy. Additionally it would be interesting however to try a fuzzy approach on this problem where planets have membership in all class labels but just to differing degrees. Given the sparsity of our knowledge about planets, their features, and habitability, a fuzzy approach may be more suitable rather than more traditional classification approaches.

Appendices

A. Proof of CES Production Function

The generalized CES production function for two parameters is

$$Q = k \cdot ((a \cdot A)^{\rho} + (b \cdot B)^{\rho})^{\eta/\rho} \tag{A.1}$$

where k>0,a+b=1, $\rho \leq 1$ and $\eta > 0$
Consider the case of DRS: where $\eta < 1$
Elasticity of substitution, $\sigma = 1/(1-\rho)$
Let the class label be y
The cost can be estimated as, c = $w_a \cdot A + w_b \cdot B$
Lagrange function,

$$\mathcal{L} = w_a \cdot A + w_b \cdot B - \lambda(f(A,B) - y); \tag{A.2}$$

Minimizing,

$$\begin{aligned} f_A =& \frac{\partial f}{\partial A} = \frac{\partial k \cdot [a \cdot A_\rho + b \cdot B_\rho]^{\eta/\rho}}{\partial A} \\ &= k \cdot \frac{\eta}{\rho}[a \cdot A_\rho + b \cdot B_\rho]^{\frac{\eta-\rho}{\rho}} \cdot a\rho \cdot A^{\rho-1} \\ &= k \cdot \eta[a \cdot A_\rho + b \cdot B_\rho]^{\frac{\eta-\rho}{\rho}} \cdot aA^{\rho-1} \\ &= m \cdot aA^{\rho-1} \end{aligned} \tag{A.3}$$

Similarly,

$$f_B = m \cdot bB^{\rho-1} \tag{A.4}$$

$$f_\lambda = \frac{\partial \mathcal{L}}{\partial \lambda} = -f(A,B) + y$$

The first order conditions are,

$$\begin{aligned} &\frac{\partial \mathcal{L}}{\partial A} = w_a - \lambda maA^{\rho-1} = 0; \\ &\Rightarrow\ w_a = \lambda maA^{\rho-1}; \\ &\frac{\partial \mathcal{L}}{\partial B} = w_b - \lambda mbB^{\rho-1} = 0; \\ &\Rightarrow\ w_b = \lambda mbB^{\rho-1}; \\ &\frac{\partial \mathcal{L}}{\partial \lambda} = 0; \\ &\Rightarrow\ f(A,B) = y; \\ &\Rightarrow\ y = k \cdot [aA^{\rho} + bB^{\rho}]^{\eta/\rho}; \end{aligned}$$

$$
\begin{aligned}
&\frac{w_b}{w_a} = \frac{\lambda mbB^{\rho-1}}{\lambda maA^{\rho-1}} \\
&\Rightarrow \frac{w_b}{w_a} = \frac{b}{a} \cdot \frac{B^{\rho-1}}{A^{\rho-1}} \\
&\Rightarrow B^{\rho-1} = \frac{w_b}{w_a} \cdot \frac{a}{b} \cdot A^{\rho-1} \\
&\Rightarrow B = [\frac{w_b}{w_a} \cdot \frac{a}{b} \cdot A^{\rho-1}]^{\frac{1}{\rho-1}} \\
&\Rightarrow B = [\frac{w_b}{w_a} \cdot \frac{a}{b}]^{\frac{1}{\rho-1}} \cdot A
\end{aligned}
$$

Since,

$$
\begin{aligned}
&\Rightarrow y = k \cdot [aA^\rho + bB^\rho]^{\eta/\rho}; \\
&\Rightarrow y = k \cdot [aA^\rho + b\frac{w_b}{w_a} \cdot \frac{a}{b}]^{\frac{\rho}{\rho-1}} \cdot A^\rho]^{\eta/\rho}; \\
&= k \cdot A^\eta [a + b(\frac{w_b}{w_a} \cdot \frac{a}{b})^{\frac{\rho}{\rho-1}}]^{\eta/\rho} \\
&\Rightarrow y \cdot k^{-1} = A^\eta [a + b(\frac{w_b}{w_a} \cdot \frac{a}{b})^{\frac{\rho}{\rho-1}}]^{\eta/\rho} \\
&\Rightarrow A^\eta = y \cdot k^{-1} [a + b(\frac{w_b}{w_a} \cdot \frac{a}{b})^{\frac{\rho}{\rho-1}}]^{-\eta/\rho} \\
&\Rightarrow A = (yk^{-1})^{1/\eta} [a + b(\frac{w_b}{w_a} \cdot \frac{a}{b})^{\frac{\rho}{\rho-1}}]^{-\frac{1}{\rho}} \\
&\Rightarrow w_a \cdot A = w_a \cdot (yk^{-1})^{1/\eta} [a + b(\frac{w_b}{w_a} \cdot \frac{a}{b})^{\frac{\rho}{\rho-1}}]^{-\frac{1}{\rho}}
\end{aligned}
$$

Similarly,

$$
\Rightarrow w_b \cdot B = w_b \cdot (yk^{-1})^{1/\eta} [a(\frac{w_a}{w_b} \cdot \frac{b}{a})^{\frac{\rho}{\rho-1}} + b]^{-\frac{1}{\rho}}
$$

The cost function,

$$
\begin{aligned}
c &= w_a A + w_b B \\
&= w_a (yk^{-1})^{1/\eta} [a + b \cdot (\frac{w_b}{w_a} \cdot \frac{a}{b})^{\frac{\rho}{\rho-1}}]^{\frac{-1}{\rho}} + w_b (yk^{-1})^{1/\eta} [a \cdot (\frac{w_a}{w_b} \cdot \frac{b}{a})^{\frac{\rho}{\rho-1}} + b]^{\frac{-1}{\rho}} \\
&= (yk^{-1})^{1/\eta} w_a \cdot [a + b \cdot (\frac{w_b}{w_a} \cdot \frac{a}{b})^{\frac{\rho}{\rho-1}}]^{\frac{-1}{\rho}} + w_b \cdot [a \cdot (\frac{w_a}{w_b} \cdot \frac{b}{a})^{\frac{\rho}{\rho-1}} + b]^{\frac{-1}{\rho}}
\end{aligned}
$$

The conditions for Optimization are:

$$
\begin{aligned}
&p\frac{\partial f}{\partial A} = w_a \\
&\Rightarrow p f_A = w_a \\
&\Rightarrow p \cdot maA^{\rho-1} = w_a
\end{aligned}
$$

Therefore, we can write

$$
\begin{aligned}
w_a &= p \cdot maA^{\rho-1} \\
w_b &= p \cdot mbB^{\rho-1}
\end{aligned}
$$

$$
\begin{aligned}
w_a &= p \cdot maA^{\rho-1} \\
&= p \cdot k \cdot \eta[aA^{\rho}bB^{\rho}]^{\frac{\eta-\rho}{\rho}} \cdot aA^{\rho-1} \\
&= p \cdot k \cdot \eta aA^{\rho-1}[aA^{\rho} + b(\frac{w_b}{w_a} \cdot \frac{a}{b})^{\frac{\rho}{\rho-1}} \cdot A^{\rho}]^{\frac{\eta-\rho}{\rho}} \\
&= p \cdot k \cdot \eta aA^{\rho-1} \cdot A^{\eta-\rho}[aA^{\rho} + b(\frac{w_b}{w_a} \cdot \frac{a}{b})^{\frac{\rho}{\rho-1}}]^{\frac{\eta-\rho}{\rho}} \\
&= p \cdot k \cdot \eta aA^{\eta-1} \cdot [a + b(\frac{w_b}{w_a} \cdot \frac{a}{b})^{\frac{\rho}{\rho-1}}]^{\frac{\eta-\rho}{\rho}} \\
&= p \cdot k \cdot \eta aA^{\eta-1} \cdot [a + b(\frac{w_b}{w_a})^{\frac{\rho}{\rho-1}} \cdot (a/b)^{\frac{\rho}{\rho-1}}]^{\frac{\eta-\rho}{\rho}} \\
&= p \cdot k \cdot \eta aA^{\eta-1} \cdot [a + b^{1-\frac{\rho}{\rho-1}} \cdot a^{\frac{\rho}{\rho-1}} \cdot (\frac{w_b}{w_a})^{\frac{\rho}{\rho-1}}]^{\frac{\eta-\rho}{\rho}} \\
\Rightarrow w_a &= p \cdot k \cdot \eta aA^{\eta-1} \cdot [a + b^{\frac{-1}{\rho-1}} \cdot a^{\frac{\rho}{\rho-1}} \cdot (\frac{w_b}{w_a})^{\frac{\rho}{\rho-1}}]^{\frac{\eta-\rho}{\rho}} \\
\Rightarrow A^{1-\eta} &= p \cdot k \cdot \eta a(w_a)^{-1} \cdot [a + b^{\frac{1}{1-\rho}} \cdot a^{\frac{\rho}{\rho-1}} \cdot (\frac{w_b}{w_a})^{\frac{\rho}{\rho-1}}]^{\frac{\eta-\rho}{\rho}} \\
\Rightarrow A &= (p \cdot k \cdot \eta a(w_a)^{-1} \cdot [a + b^{\frac{1}{1-\rho}} \cdot a^{\frac{\rho}{\rho-1}} \cdot (\frac{w_b}{w_a})^{\frac{\rho}{\rho-1}}]^{\frac{\eta-\rho}{\rho}})^{\frac{1}{1-\eta}}
\end{aligned}
$$

$$
\begin{aligned}
\Rightarrow w_a &= p \cdot k \cdot \eta aA^{\eta-1} \cdot [a + b \cdot (\frac{w_b}{w_a} \cdot \frac{a}{b})^{\frac{\rho}{\rho-1}}]^{\frac{\eta-\rho}{\rho}} \\
\Rightarrow A^{1-\eta} &= w_a^{-1} p \cdot k \cdot \eta a \cdot [a + b \cdot (\frac{w_b}{w_a} \cdot \frac{a}{b})^{\frac{\rho}{\rho-1}}]^{\frac{\eta-\rho}{\rho}} \\
\Rightarrow A &= [w_a^{-1} p \cdot k \cdot \eta a \cdot [a + b \cdot (\frac{w_b}{w_a} \cdot \frac{a}{b})^{\frac{\rho}{\rho-1}}]^{\frac{\eta-\rho}{\rho}}]^{\frac{1}{1-\eta}} \\
\Rightarrow A &= [w_a^{-1} p \cdot k \cdot \eta a]^{\frac{1}{1-\eta}} \cdot [a + b \cdot (\frac{w_b}{w_a} \cdot \frac{a}{b})^{\frac{\rho}{\rho-1}}]^{\frac{\eta-\rho}{\rho \cdot (1-\tan)}}
\end{aligned}
$$

Similarly, we can write

$$
\Rightarrow B = [w_b^{-1} p \cdot k \cdot \eta b]^{\frac{1}{1-\eta}} \cdot [a \cdot (\frac{w_a}{w_b} \cdot \frac{b}{a})^{\frac{\rho}{\rho-1}}]^{\frac{\eta-\rho}{\rho \cdot (1-\eta)}}
$$

Since,

$$
\begin{aligned}
Y &= [aA^{\rho} + bB^{\rho}]^{\eta/\rho} \qquad \text{(A.5)} \\
A^{\rho} &= [w_a^{-1} p \cdot k \cdot \eta a]^{\frac{\rho}{1-\eta}} \cdot [a + b \cdot (\frac{w_b}{w_a} \cdot \frac{a}{b})^{\frac{\rho}{\rho-1}}]^{\frac{\eta-\rho}{(1-\eta)}} \\
a \cdot A^{\rho} &= a^{1+\frac{\rho}{1-\eta}} [w_a^{-1} p \cdot k \cdot \eta]^{\frac{\rho}{1-\eta}} \cdot [a + b \cdot (\frac{w_b}{w_a} \cdot \frac{a}{b})^{\frac{\rho}{\rho-1}}]^{\frac{\eta-\rho}{(1-\eta)}}
\end{aligned}
$$

$$
\begin{aligned}
&and, \qquad \text{(A.6)} \\
b \cdot B^{\rho} &= b^{1+\frac{\rho}{1-\eta}} [w_b^{-1} p \cdot k \cdot \eta]^{\frac{\rho}{1-\eta}} \cdot [b \cdot (\frac{w_a}{w_b} \cdot \frac{b}{a})^{\frac{\rho}{\rho-1}}]^{\frac{\eta-\rho}{(1-\eta)}}
\end{aligned}
$$

Replace the values of $a.A^{\rho}$ and $b.B^{\rho}$ in A.5,

$$
\begin{aligned}
Y &= [a^{\frac{1+\rho-\eta}{1-\eta}} \cdot (w_a^{-1} p \cdot k \cdot \eta)^{\frac{\rho}{1-\eta}} \cdot [a + b \cdot (\frac{w_b}{w_a} \cdot \frac{a}{b})^{\frac{\rho}{\rho-1}}]^{\frac{\eta-\rho}{1-\eta}} + b^{\frac{1+\rho-\eta}{1-\eta}} (w_b^{-1} p \cdot k \cdot \eta)^{\frac{\rho}{1-\eta}} \cdot [a \cdot (\frac{w_a}{w_b} \cdot \frac{b}{a})^{\frac{\rho}{\rho-1}} + b]^{\frac{\eta-\rho}{1-\eta}}]^{\eta/\rho} \\
\Rightarrow Y &= (p \cdot k \cdot \eta)^{\frac{\eta}{1-\eta}} \cdot [a^{\frac{1+\rho-\eta}{1-\eta}} \cdot w_a^{\frac{\rho}{\eta-1}} \cdot [a + b(\frac{w_b}{w_a} \cdot \frac{a}{b})^{\frac{\rho}{\rho-1}}]^{\frac{\eta-\rho}{1-\eta}} + b^{\frac{1+\rho-\eta}{1-\eta}} \cdot w_b^{\frac{rho}{\eta-1}} \cdot [a(\frac{w_a}{w_b} \cdot \frac{b}{a})^{\frac{\rho}{\rho-1}}]^{\frac{\eta-\rho}{1-\eta}}]^{\eta/\rho}
\end{aligned}
$$

There are 3 ranges of values of ρ and η

$$\eta < 0$$
$$0 \leq \eta < 1$$
$$1 \leq \eta$$

Similarly,

$$\rho < 0$$
$$0 \leq \rho < 1$$
$$1 \leq \rho$$

For DRS, the following condition should satisfy,

$$0 < \eta < 1$$

Therefore, we can write that for $0 < \eta < 1$, the value of ρ can be

$$\rho < 0$$
$$0 \leq \rho < 1$$
$$1 \leq \rho$$

B. Proof of Model Scalability

Here we prove optimality using Hessian matrices. If a Hessian matrix of a function is symmetric about its primary diagonal, a global optimum exists for that function. The general form of a Hessian matrix for a function is given by:

$$\text{Hess}\,(Y) = \begin{bmatrix} \frac{\partial^2 Y}{\partial A^2} & \frac{\partial^2 Y}{\partial B \partial A} \\ \frac{\partial^2 Y}{\partial A \partial B} & \frac{\partial^2 Y}{\partial B^2} \end{bmatrix} \tag{B.1}$$

Here, the elements of $\text{Hess}\,(Y)$ are given as:

$$\begin{aligned} \frac{\partial^2 Y}{\partial A^2} &= ka\eta\Big[(\rho-1)A^{\rho-2}(aA^\rho + bB^\rho)^{\frac{\eta-\rho}{\rho}} + \frac{\eta-\rho}{\rho}A^{\rho-1}(aA^\rho + bB^\rho)^{\frac{\eta-2\rho}{\rho}}\Big] \\ \frac{\partial^2 Y}{\partial B \partial A} &= kab\eta(\eta-\rho)A^{\rho-1}B^{\rho-1}(aA^\rho + bB^\rho)^{\frac{\eta-2\rho}{\rho}} \\ \frac{\partial^2 Y}{\partial A \partial B} &= kab\eta(\eta-\rho)A^{\rho-1}B^{\rho-1}(aA^\rho + bB^\rho)^{\frac{\eta-2\rho}{\rho}} \\ \frac{\partial^2 Y}{\partial B^2} &= ka\eta\Big[(\rho-1)B^{\rho-2}(aA^\rho + bB^\rho)^{\frac{\eta-\rho}{\rho}} + \frac{\eta-\rho}{\rho}B^{\rho-1}(aA^\rho + bB^\rho)^{\frac{\eta-2\rho}{\rho}}\Big] \end{aligned} \tag{B.2}$$

From the Equations B.2, we can see that:

$$\frac{\partial^2 Y}{\partial B \partial A} = \frac{\partial^2 Y}{\partial A \partial B}$$

This implies that $\text{Hess}\,(Y)$ is symmetric about the primary diagonal, and hence, Y has a global optimum.

For a CES production function with n terms, the general form the the elements of $\mathrm{Hess}\,(Y)$ is given as:

If $i = j$,

$$a_{ij} = k\eta\alpha_i \left[(\rho - 1) A_i^{\rho-2} \Big(\sum_{m=1}^{n} \alpha_m A_m^{\rho} \Big)^{\frac{\eta-\rho}{\rho}} + \frac{(\eta - \rho)}{\rho} A_i^{\rho-1} \Big(\sum_{m=1}^{n} \alpha_m A_m^{\rho} \Big)^{\frac{\eta-2\rho}{\rho}} \right] \tag{B.3}$$

If $i \neq j$,

$$a_{ij} = k\eta(\eta - \rho)\alpha_i \alpha_j A_i^{\rho-1} A_j^{\rho-1} \Big(\sum_{m=1}^{n} \alpha_m A_m^{\rho} \Big)^{\frac{\eta-2\rho}{\rho}} \tag{B.4}$$

$\forall 1 \leq i \leq n, 1 \leq j \leq n$; α_i is the i^{th} coefficient and A_i is the i^{th} parameter. For any $n \in \{1, 2, 3, ...\}$, the element in the $(i, j)^{th}$ position of the Hessian matrix is given by Equations B.3 and B.4. From Equation B.4, it is evident that all of the non-diagonal elements are symmetric about (i, j). Hence, the Hessian matrix of a CES production function with any number of variables is always symmetric about the primary diagonal.

Thus, we conclude that the CES function has a global optimum, and is scalable for any $n \in \{1, 2, 3, ...\}$.

C. MATLAB Codes

Here we present Matlab codes that implement the analytical model, compute the scores for the entire dataset.

Function fmincon

The function *fmincon* finds a constrained minimum of a scalar function of multivariable starting at an initial point. This is generally known as constrained nonlinear optimization. Function *fmincon* solves problems of the form:

min $f(x)$ subject to x,

$$\begin{cases} A * x \leq b \\ A_{eq} * x = b_{eq} \end{cases}$$

are the linear constraints, and the following equations are the non-linear constraints:

$$\begin{cases} C * x \leq 0 \\ C_{eq} * x = 0 \end{cases}$$

are the linear constraints, and the following equations are the non-linear constraints:

$$\begin{cases} C * x \leq 0 \\ C_{eq} * x = 0 \end{cases}$$

and bounding of variables

$$\begin{cases} lb \leq x \\ x \leq ub \end{cases}$$

This has been applied to the cases **CRS** and **DRS** for the CEESA and CES score computation.

C.1. ***Constant Returns to Scale***

Applying the constraints:

$$\begin{cases} a + b + c + d + e = 1 \\ \rho \leq 1,\ \nu = 1 \end{cases}$$

to the function: $Y = (a.x_1^{\rho} + b.x_2^{\rho} + c.x_3^{\rho} + d.x_4^{\rho} + e.x_5^{\rho})^{\nu/\rho}$; use *fmincon* to compute ρ and ν for optimum Y.

C.2. ***Decreasing Returns to Scale***

Applying the constraints:

$$\begin{cases} a+b+c+d+e=1 \\ \rho \leq 1,\ \nu < 1 \end{cases}$$

to the function: $Y = (a.x_1^{\rho} + b.x_2^{\rho} + c.x_3^{\rho} + d.x_4^{\rho} + e.x_5^{\rho})^{\nu/\rho}$; use *fmincon* to compute ρ and ν for optimum Y.

C.3. ***Working of fmincon***

[x,fval] = fmincon(fun,x_0, A, b) starts at point x_0 and finds a minimum x to the function described in fun subject to the linear inequalities, $A * x \leq b$, where A is a matrix, x and b are vectors and x_0 can be a scalar, a vector or a matrix. It also returns the value of the objective function **fun** at the solution x.

[x,fval] = fmincon(fun,$x_0, A, b, A_{eq}, b_{eq}$) starts at x_0 and minimizes **fun** subject to the linear inequalities $A_{eq} * x = b_{eq}$ and $A * x \leq b$, where A_{eq} is a matrix and b_{eq} is a vector. It also returns the value of the objective function **fun** at the solution x.

[x,fval] = fmincon(fun,$x0, A, b, A_{eq}, b_{eq}, lb, ub$) defines a set of lower and upper bounds on the design variables in x, so that the solution is always in the range $lb \leq x \leq ub$. If no equalities exist, set Aeq = [] and beq = []. If x(i) is unbounded below, set lb(i) = -Inf, and if x(i) is unbounded above, set ub(i) = Inf (fmincon, 2017).

References

Abazajian K. N. et al., The Seventh Data Release of the Sloan Digital Sky Survey, Astrophysical Journal Supplement 182 (2009) 543-558, doi:10.1088/0067-0049/182/2/543

Strigari, L. E., Barnabè, M., Marshall, P. J., & Blandford, R. D., 2012. Nomads of the Galaxy. *Mon. Not. R. Astron. Soc.* , 423, 1856

Schulze-Makuch, D., Méndez, A., Fairén, A. G., et al., 2011. A Two-Tiered Approach to Assessing the Habitability of Exoplanets. *Astrobiology*, 11, 1041

Batalha, N. M., 2014. Exploring exoplanet populations with NASA's Kepler Mission. *Proceedings of the National Academy of Science*, 111, 12647

Irwin, L.N. & Schulze-Makuch, D., 2011. Cosmic Biology: How Life Could Evolve on Other World. Springer-Praxis, New York.

Irwin, L. N., Méndez, A., Fairén, A. G., & Schulze-Makuch, D., 2014. Assessing the Possibility of Biological Complexity on Other Worlds, with an Estimate of the Occurrence of Complex Life in the Milky Way Galaxy. *Challenges*, 5, 159

Bora, K., Saha, S., Agrawal, S., Safonova, M., Routh, S., Narasimhamurthy, A., 2016. CD-HPF: New Habitability Score via Data Analytic Modeling. *Astronomy and Computing*, 17, 129

Heller, R., & Armstrong, J., 2014. Superhabitable worlds. *Astrobiology*, 14, 50

Méndez A., 2011, A Thermal Planetary Habitability Classification for Exoplanets, University of Puerto Rico at Arecibo, url:http://phl.upr.edu/library/notes/athermalplanetaryhabitabilityclassi ficationforexoplanets

Swift J. J., Johnson J. A., Morton T. D., Crepp J. R., Montet B. T., Fabrycky D. C., Muirhead P. S., 2013, Characterizing the Cool KOIs IV: Kepler-32 as a prototype for the formation of compact planetary systems throughout the Galaxy, *Astrophys. J.*, Vol. 764, Number 1, p. 105, doi:10.1088/0004-637X/764/1/105

Kaltenegger L, Udry S., Pepe F, A Habitable Planet around HD 85512, preprint(arXiv:1108.3561)

Hájková, D. and Hurnik, J., 2007. Cobb-Douglas: The Case of a Converging Economy, *Czech Journal of Economics and Finance (Finance a uver)*, 57, 465

Nemirovski, Arkadi S., and Todd, M. J., 2008. Interior-point methods for optimization. *Acta Numerica*, 17, 191. doi:10.1017/S0962492906370018.

Elmer G. Wiens, Production Functions, url:http://www.egwald.ca/economics/cesproductionfunctions.php, Retrieved on 10/31/2017.

Hawkin Qian, Constant elasticity of substitution function and its properties, url://http://www.hawkinqian.com/blog/86/, Retrieved on 10/31/2017.

Documentation on fmincon, url://http://www.in.mathworks.com, Retrieved on 12/04/2017.

Isaac Asimov, 1988, The relativity of wrong, Essays on Science.